Design of Coded Slotted ALOHA with Interference Cancellation Errors

Charles Dumas, Lou Salaün, Iman Hmedoush, Cédric Adjih, Chung Shue Chen

▶ **To cite this version:**

Charles Dumas, Lou Salaün, Iman Hmedoush, Cédric Adjih, Chung Shue Chen. Design of Coded Slotted ALOHA with Interference Cancellation Errors. IEEE Transactions on Vehicular Technology, 2021. ⟨hal-03266615v2⟩

HAL Id: hal-03266615

https://inria.hal.science/hal-03266615v2

Submitted on 18 Jan 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HAL Authorization

Design of Coded Slotted ALOHA with Interference Cancellation Errors

Charles Dumas, Lou Salaün, *Member, IEEE*, Iman Hmedoush, *Student Member, IEEE*, Cédric Adjih, *Member, IEEE*, and Chung Shue Chen *Senior Member, IEEE*

***Abstract*—Coded Slotted ALOHA (CSA) is a random access scheme based on the application of packet erasure correcting codes to transmitted packets and the use of successive interference cancellation at the receiver. CSA has been widely studied and a common assumption is that interference cancellation can always be applied perfectly. In this paper, we study the design of CSA protocol, accounting for a non-zero probability of error due to imperfect interference cancellation (IC). A classical method to evaluate the performance of such protocols is density evolution, originating from coding theory, and that we adapt to our assumptions. Analyzing the convergence of density evolution in asymptotic conditions, we derive the optimal parameters of CSA, i.e., the set of code selection probabilities of users that maximizes the channel load. A new parameter is introduced to model the packet loss rate of the system, which is non-zero due to potential IC errors. Multi-packet reception (MPR) and the performance of 2-MPR are also studied. We investigate the trade-off between optimal load and packet loss rate, which sheds light on new optimal distributions that outperform known ones. Finally, we show that our asymptotic analytical results are consistent with simulations obtained on a finite number of slots.**

***Index Terms*—Coded slotted ALOHA, random access, interference cancellation, decoding failure, SIC failure, density evolution.**

I. Introduction

Random multiple access protocols have been widely employed in wireless telecommunications to accommodate a massive number of users, with a low level of coordination between them [1], [2]. The different users can then freely join and leave the channel without interaction and regardless of the channel state. The drawback is that the interference caused by transmission collision needs to be considered, since it may lead to packet loss at the receiver side. Dealing with packet contention is one of the basic problems in multiple access schemes. The ALOHA protocol was first proposed with users sending one data block of information, i.e., a packet, when it is generated without any prior time synchronization between the users [3], [4]. When packets collide and cannot be retrieved at the receiver side, one may need to re-transmit and system stability becomes a primary issue. Meanwhile, slotted ALOHA was introduced by defining slots for packets transmissions in order to avoid partial packet collision which often occur in ALOHA and to improve the transmission efficiency.

One enhancement of ALOHA is introduced in the work of contention resolution diversity slotted ALOHA system (CRDSA) [5], where two identical copies of the data packets are transmitted in two different slots. In other terms, from a coding perspective, the data block of each user is encoded according to a $(2, 1)$-repetition code. Thanks to the repetition, the correct reception of one packet out of the two would be sufficient to decode the data block and retrieve all the information from a user. But the key performance improvement of CRDSA comes from the added use of successive interference cancellation (SIC): when a packet can be retrieved, the interference of this packet in its slots (including the slot where the other copy was transmitted) is subtracted. It is shown that the iterative packet decoding of the copies and the use of SIC would allow for a huge improvement in the system throughput performance and enable more users to share the channel.

Irregular Repetition Slotted ALOHA (IRSA) is proposed by [6] to improve the performance of CRDSA. Instead of repeating the user data packets a predefined number of times (namely 2 for CRDSA), the idea is to let each user choose its number of repetitions according to a *probability distribution*, which is subject to optimization. One interesting finding is that for asymptotically long frames, if the number of users does not exceed a certain load threshold, then the packets of all users can be retrieved without loss. The resulting system throughput of IRSA is much higher than that of CRDSA, and can achieve asymptotically the limit of one retrieved packet per slot [7]. By strong analogy with the message-passing decoding process of Low-Density Parity-Check codes [8]–[10], the performance of IRSA can also be obtained through the method of *density evolution*, as pioneered by [6]. The system can then be optimized by deriving a formal optimization problem including constraints expressing the convergence of this density evolution.

Coded Slotted ALOHA (CSA) system [11] is introduced to generalize IRSA. It splits each data block of a user into several packets before applying a packet-oriented linear block code yielding coded packets, instead of simply using a repetition code like IRSA. The linear block code is selected from a predefined set, according to a *code probability distribution*. The *code rate* is defined as the inverse of the average number of packets repetitions for IRSA and as the average code rate of the linear block codes for CSA. Without any constraint on the code rate, IRSA achieves performance as good as CSA in terms of throughput. However, the resulting energy efficiency of the system is much lower with low rate codes. A minimal bound on the code rate is therefore often added to take into account this energy efficiency. CSA provides then an added value by outperforming IRSA for medium rate codes: between code rates $1/3$ and $1/2$. The optimization of CSA resorts to similar techniques as IRSA.

This work is supported by INRIA-Nokia-Bell-Labs Joint Research Center.

C. Dumas is with Centrale-Supelec, Gif-sur-Yvette 91190, France. He was with Nokia Bell Labs, Paris-Saclay. Email: charles.dumas@supelec.fr.

C. S. Chen and L. Salaün are with Nokia Bell Labs, Paris-Saclay, Nozay 91620, France. Email: {chung_shue.chen, lou.salaun}@nokia-bell-labs.com.

C. Adjih and I. Hmedoush and are with INRIA Saclay, Palaiseau 91120, France. Email: {cedric.adjih, iman.hmedoush}@inria.fr.

Another setting is to consider multi-packet reception (MPR) [12], namely K-MPR, where $K \geq 1$ is the maximal number of packets that can be simultaneously decoded in a slot [13], [14]. In this more general model, it is considered that one can instantly decode at most $k \leq K$ colliding packets in each slot and subtract their interference from the collision. K is a fixed parameter and a system model without multi-packet reception corresponds to $K = 1$. Note that K-MPR would allow significant throughput gain, but from an information-theoretic capacity perspective, one would expect that it requires an equivalent decrease in capacity by a factor K or greater, for example of the packet payload. A question is whether K-MPR can asymptotically achieve the maximal throughput of K packets per slot, since it has been numerically shown to approach it closely with specific schemes [15], but it has also been proven that classical IRSA does not reach this value asymptotically for any $K \geq 2$ [16].

CSA over packet erasure channel and block erasure channel has been studied and optimized in [17]. When modeling (e.g., with density evolution), a common assumption in the CSA literature is that interference cancellation (IC) can always be applied perfectly and additionally that a singleton packet can be successfully decoded with probability 1. A transmitted packet is denoted a *singleton packet* when there are no packets left from other users that were transmitted in the same slot, possibly after having performed some interference cancellations. In reality, in practical communication systems and signal processing under noise and imperfection, the probability of interference cancellation error is in general non-zero. Simple extensions of density evolution that consider errors were also proposed in [6] but were not exploited, neither in an optimization problem nor through numerical results. In this paper, we will consider a similar model, in which the probability of correctly retrieving a packet decreases after every interference cancellation applied to the packets in a slot. One can see that in this new configuration, the optimal code probability distributions may be affected and lead to completely different load thresholds. It is therefore important to study CSA under this model in order to design optimal CSA protocols for real systems in which interference cancellation errors generally exist.

Moreover, since all packets of a user can be lost due to interference cancellation failure, there is a non-zero probability that a user's information is completely lost at the end of the iterative process. For this reason, similarly to [17], a user recovery ratio α will be introduced and used as a new parameter for the system in our analysis. The load of the system will be optimized with regards to the desired user recovery ratio.

The main contributions of this paper are summarized below:

1) We investigate CSA protocol under the new model which takes into account packet retrieval errors caused by interference cancellation (IC). We provide the new optimal code probability distributions under this generalized model for maximizing the system load with respect to desired user recovery ratio. Besides, we investigate the trade-off between the optimal load and the loss rate.
2) The above optimal code probability distributions are determined by linear programming for 1-MPR systems. The 2-MPR system load maximization problem is non-linear. We provide a sharp linear approximation to derive the optimal code probability distributions in 2-MPR system. This approximation is novel and the method can be generally reused for similar optimization problems.
3) Some numerically optimized distributions under the assumption of zero IC error were given in [6], [11] while a family of truncated Soliton distributions has been shown optimal asymptotically [7]. In this paper, we show that such distributions produce unsatisfactory performance under the generalized model with IC error, which our derived new distributions outperform.
4) We conduct a complete performance evaluation and comprehensive simulation to study the suitability of the obtained analytical results for practical case of finite length frames. This paves the way for realistic implementation and application in practical systems.

The rest of the paper is organized as follows. In Section II, the system model with IC errors is specified and different system parameters and performance metrics are introduced. The bipartite graph representation and the study of density evolution are presented in Section III, with a focus on 2-MPR in Section IV. In Section V, CSA parameters are studied and optimized. We show the trade-off between the error rate and system load. Simulation results are presented in Section VI and compared with the analytical results. Finally, conclusions about optimal distributions are drawn in Section VII.

II. System Model

A. Principle of Coded Slotted ALOHA

We consider a random access scheme where M users transmit data to a common receiver, e.g., a base station, via a shared channel composed of frames. Each frame is divided into kN slots. Each user transmits one data block per Medium Access Control (MAC) frame and all transmissions are slot synchronized. In this paper, we study the phenomena occurring on a single MAC frame. Information is divided and contained in packets, with each packet being transmitted in a random slot. In each slot, the base station receives the superposition of all transmitted packets. A packet can be instantly *retrieved* if it is the sole packet transmitted in its slot[1]. However, if several users transmit in the same slot, a collision arises and no packet can be retrieved yet. Without SIC, the packets would be lost and transmitted again by the same users in a future frame.

The above random access protocol is the framework for both IRSA [6], and more general CSA [11]. In each frame, the data block of each user is first encoded into a number of (coded) packets, through simple repetition for IRSA, or through a linear block code for CSA. We explained that a packet arriving alone at a slot can be correctly retrieved. Thanks to pointers and a preamble attached to each packet,

[1]In the remaining of the article, we use the terms "retrieved" when the symbols of the packet are retrieved from the physical signal, e.g., including demodulation, and error correction with channel codes; we reserve the term "decoded" for decoding of the linear block code of CSA.

the time positions of the other (coded) packets transmitted by the user are available. If enough packets of the data block of one user have already been retrieved, maximum-a-posteriori (MAP) erasure decoding is performed to decode all the others. Then, all packets associated to this data block can be reconstructed, and the interference on their corresponding slots can be mitigated by SIC techniques. Due to retrieved packets and SIC, some packets of other users may in turn become collision-free in their slots, i.e., become *singletons*. The iteration of MAP erasure decoding and SIC may make possible to correctly receive the data blocks from all users. This will be referred as the *iterative process*. Some variants of IRSA, such as network-coded slotted ALOHA [18], shifted coded slotted ALOHA [19] or ZigZag decodable coded slotted ALOHA [20], do not rely on the retrieval of singletons, but they require a more complex decoder, and are not considered here.

In the CSA protocol, the user's information is first split into a fixed number of k packets and then encoded following a linear block code $C_i(n_i, k)$. At the end of the encoding process, the information is therefore split and encoded into n_i packets. With a (n_i, k) code, the correct retrieval of k packets is sufficient to recover the complete information sent by a user. The total number of slots in a frame is kN.

Each of the n_i encoded packets is then transmitted in a single slot chosen randomly. In the general case of CSA, the codes C_i are chosen over a predefined set of codes selected by the protocol designer. In this paper, we only consider the use of maximum distance separable codes (MDS) [21], i.e., a set of codes $(k+1, k), (k+2, k), \ldots, (L, k)$, where $L > k$ is a parameter of the system representing the maximal number of coded packets that can be obtained after encoding the data block of one user, i.e., the *maximum code length*. It is also directly linked to the minimal local rate of the system, defined by the fraction k/L. Note that it is important to optimize the parameter L since by its connection to the rate, thus the average number of transmissions, it is also linked to the energy consumption of the system. For a fixed k, we write k-CSA for the corresponding CSA system. The main parameter of the system is how users are selecting one of the linear block codes C_i, which is at random, and is modeled by a code probability distribution Λ. When considering k-CSA, Λ_i is the probability to choose MDS code (i, k), where $i > k$. Fig. 1 depicts a small 2-CSA system with 3 users transmitting during a frame composed of 8 slots. The (coded) packets sent by each user are represented by colored rectangles, and they are sent in the same wireless channel. Initially each user has one data block that is first split in 2 initial (uncoded) packets. The probability to select code C_3 (which is a [3,2]-linear block code, e.g., parity check code) is $\Lambda_3 = \frac{1}{3}$, and the probability to select code C_4 (which is a [4,2]-linear block code) is $\Lambda_4 = \frac{2}{3}$. User 1 and user 3 randomly select code C_4, thus they encode their 2 initial (uncoded) packets into 4 (coded) packets that are then sent on random slots, here, on slots $1, 2, 3$ and 6 for user 1, and slots $4, 5, 6, 7$ for user 3. User 2 randomly selects code C_3, thus encodes its 2 initial (uncoded) packets into 3 (coded) packets that are sent in slots $3, 4$ and 5.

B. Successive Interference Cancellation Model

A common assumption in the results of the CSA literature is that successive interference cancellation can always be applied perfectly such that a packet can be correctly retrieved by eliminating the interference due to collided packets. In this paper, we will study the impact of SIC error and optimize the design of CSA under this more realistic and general model. Note that the succession of interference subtractions may lead to the wrong retrieval of a packet. A wrongly retrieved packet would result in a packet loss and cannot be used to subtract interference for other remaining packets on the slot.

Let ℓ be the number of packets collided in a slot. We denote by w_ℓ the probability that a packet can be correctly retrieved after SIC (i.e., by eliminating the other $\ell-1$ packets collided in the slot). In this paper, we consider the case of an exponential error due to SIC:

$$w_\ell \triangleq \gamma^{\ell-1}, \tag{1}$$

where γ is defined as the *interference cancellation efficiency* (or *SIC efficiency*) for our model. γ is the probability to correctly cancel the interference of one packet on one slot and this probability is independent of the other packets in the slot. Note that $\gamma = 1$ would lead to the classical model of SIC without error studied in [11], which is a special case of (1). Note also that conversely [6, Appendix A] proposes a short generalization of the density evolution for IRSA for the case where the ℓ^{th} packet is retrieved with an arbitrary probability w_ℓ: when applied to the special case of IRSA (instead of our more general case of CSA), and for 1-MPR, our later equations can also be derived from [6, Eqn. (9)].

The above loss model (1) is an idealized model of the SIC process. A more accurate modeling of the physical layer inter-slot SIC process has been described in [6]. In particular, before any SIC operation, the signal parameters (amplitude, phase, frequency offset) might need to be estimated. Estimation errors could happen and they lead to incorrect signal subtraction (cancellation). After each SIC operation, residual energy is present due to the imperfection of the cancellation process. The residual energy would accumulate as the number of SIC operations on the same slot increases. Estimation errors and residual energy due to imperfect cancellation could lead to significant SIC failure probability. [6, Appendix B] and [11, Appendix A] observe that under their respective physical layer models and assumptions, the packet error rate is 10^{-2} and 10^{-3} respectively after $\ell - 1 = 7$ SIC operations, and this has been a reasonable justification for the assumption of perfect SIC in some of the literature. However, for instance, operating at lower E_b/N_0 (see [6, Fig. 9] or [11, Fig. 10]), or other reasons, would lead to an increase of those probabilities.

Our simplified model in (1), is essentially, a model where each SIC operation has the same probability $1 - \gamma$ to be the breaking-point after which the residual packets can no longer be recovered. In other words, $1 - \gamma$ is the probability that a packet cannot be correctly subtracted in a slot when SIC is applied. As a consequence, the probability to retrieve a packet on a certain slot, through canceling the interference from the other $\ell - 1$ colliding packets, is given by $\gamma^{\ell-1}$. This explains the above expression of exponential error. This model also

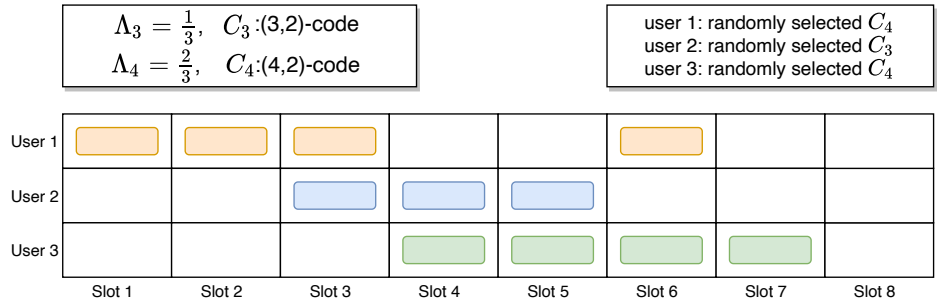

Fig. 1. Example of the CSA access scheme for $k = 2, L = 4$, with 3 users and $2 \times 4 = 8$ slots.

corresponds to an approximation where γ would be selected to be the average SIC efficiency observed from a real SIC model. In practice, γ is not expected to be constant with ℓ. For instance, considering the data points of the curve from [6, Fig. 9] for $E_b/N_0 = 1.5$ dB (resp. $E_b/N_0 = 1.8$ dB), one can observe a variation of the estimated value[2] of $1-\gamma$ of more than 40%. This is an idealized physical layer simulation, and one might expect greater variations with real hardware. On the other hand, notice that an increase of γ after a large number of SICs might also be mitigated by the fact that most packets are recovered after a limited number of SICs (typically less than 5), and that the effect of SIC errors in the later iterations of the decoding process also needs to be assessed.

Some other prior works have accurately studied the impact of errors in IRSA decoding. For instance, the impact of imperfect CSI with PDMA/IRSA [22], or equivalently estimation errors with MARSALA/CRDSA [23], [24]. Although sophisticated parameter estimation methods are used, and mathematical expressions are obtained, they do not easily translate into a simple model of SIC failure probability, as they have to be iterated in a manner that parallels the IRSA/CRDSA decoding process, and the results are obtained by simulations. Alternately, some insights could be gained from surveys, for instance, one survey of SIC for OFDM [25] and one survey of Power-Domain NOMA [26] which relies also on SIC. But even for the richer NOMA literature, the latter noted that "Although there exist works that analytically study the SIC error propagation in basic MIMO systems, there is no prominent research that provides a mathematical understanding of the effect of imperfect SIC on NOMA schemes". We reiterate their conclusion that this would constitute an interesting research direction, and in this article, we focus on our idealized SIC failure model Eqn. (1) as a first approximation that allows us to still capture the effects of SIC failures.

C. CSA Performance Metrics and Objectives

Our goal is to maximize the number of users in the channel while minimizing the amount of packet loss. To assess the efficiency of the different parameters, several performance metrics are introduced. $G \triangleq \frac{M}{N}$ is the load of the system, in other words the number of users per slot, which is to be maximized. Let S be the number of decoded users at the end of the iterative process, where $S \leq M$. Then, $T \triangleq \frac{S}{N}$ is called the throughput of the system. T is always less than or equal to G and $T = G$ if and only if all users have been correctly decoded. Finally, we have the packet loss rate (PLR) of the system $P_L \triangleq 1 - \frac{S}{M}$ which is the proportion of non-decoded users at the end of the iterative process, which we aim to minimize. It is clear that we can also write $P_L = 1 - \frac{T}{G}$.

Since all packets of a user may be lost due to SIC failure, there is a non-zero probability that a user's information is completely lost at the end of the iterative process. This has been observed in the literature studying IRSA or CSA in presence of channel losses, see [17], [27], [28] for instance. Thus, unlike IRSA or CSA without losses, there no longer exists a proper *load threshold* G^* below which all users' data blocks are retrieved with probability converging to 1. To remedy for that, [28] defines the *asymptotic decoding threshold*, and equivalently [17] defines the *user recovery ratio*, that both help redefine the idea of load threshold.

Following [17], the user recovery ratio α is also introduced in our system model, a new parameter to be addressed. α is defined as the target ratio of decoded users at the end of the process, or in other terms the wanted (minimum) probability that a user's data block is decoded successfully. Clearly, $1-\alpha$ is then the ratio of non-decoded users, i.e., the PLR. We then define a load threshold G^*_α, which is the maximum load such that a maximum ratio of $1-\alpha$ users are non-decoded. Mathematically, we define:

$$G^*_\alpha(\Lambda, \gamma) \triangleq \max \left\{ G : 1 - \frac{T(\Lambda, G)}{G} \leq 1 - \alpha \right\}, \qquad (2)$$

where γ is fixed. Λ is the main parameter to optimize in order to achieve the highest G^*_α, and CSA optimization consists in solving the following optimization problem:

Problem 1: Given $\alpha \in (0,1)$, $\gamma \in (0,1)$, $k > 0$, $L > k$, and a set of linear block codes $(C_i(i,k))_{i=k+1,\ldots,L}$, find a code distribution Λ^* that maximizes $G^*_\alpha(\Lambda, \gamma)$, that is:

$$\Lambda^* = \arg\max_{\Lambda} G^*_\alpha(\Lambda, \gamma) \qquad (3)$$

corresponding to [17, Eqn. (4) and Eqn. (5)].

Notice that the system described above will be analytically studied when $N \to \infty$ at the first place. We will then compare

[2]By denoting PER_j the Packet Error Rate (PER) after j SIC operations, and writing $\mathrm{PER}_j = 1 - \gamma_j^j(1 - \mathrm{PER}_0)$, one can compute an estimate of γ_j from the data point for $j+1$ bursts in the curve of [6, Fig. 9]; with PER_0 being the PER for the AWGN channel without SIC. For $E_b/N_0 = 1.5$ dB (resp. $E_b/N_0 = 1.8$ dB), one obtains the estimates: $\gamma_1 = 1 - 0.9 \times 10^{-2}, \gamma_3 = 1 - 1.1 \times 10^{-2}, \gamma_7 = 1 - 0.8 \times 10^{-2}$ (resp. $\gamma_1 = 1 - 1.2 \times 10^{-4}, \gamma_3 = 1 - 1.7 \times 10^{-4}, \gamma_7 = 1 - 1.6 \times 10^{-4}$).

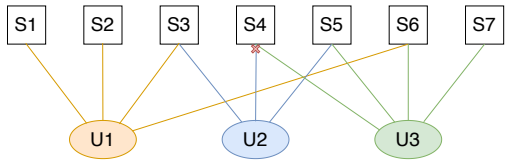

Fig. 2. Graph representation of the 2-CSA example in Fig. 1.

the derived analytic results to simulation results (in Section VI) which are obtained in a finite setting, where $N = 1000$.

D. Variants of CSA with K-MPR

An extension of CSA that will also be studied here is K-MPR [13], where K is an integer greater than or equal to 1. The difference to classical ALOHA is that K-MPR allows to retrieve instantly ℓ colliding packets in a single slot, as long as $\ell \leq K$. One can see that K-MPR would thus have higher system throughput. However, retrieving K colliding packets for $K \geq 2$ has a technological cost of implementation and/or in capacity cost, and generally considered in the literature as, at least, linearly proportional to K.[3] The normalized load G/K and normalized throughput T/K are considered to take this cost into account. Since the analytical calculations become more and more complicated as K increases, in this paper we only consider the cases where $K = 1$ and $K = 2$.

Table I summarizes the main system parameters and variables used in this paper.

III. Density Evolution Analysis

A. Bipartite Graph

By analogy with the *Tanner graphs* of coding theory [10], the CSA system can be modeled as a bipartite graph $g = (B, S, E)$ [6], with a set B of M user nodes, a set S of kN slot nodes, and the set of edges E. An edge from E is drawn between a user i and a slot j if i transmits a packet in slot j.

For the 2-CSA example in Fig. 1, Fig. 2 presents a visualization of the associated bipartite graph, with the user nodes drawn as ellipses and the slot nodes drawn as squares. In this example, user 1 is the only user transmitting in slots 1 and 2, so these singleton packets can be instantly retrieved. Since user 1 used a (4,2)-code and two coded packets are known, its data block can be decoded, and then the third coded packet can be deduced as well. By eliminating the transmissions of user 1 in slots 1, 2, 3 and 6, now two of the coded packets of users 3 can be retrieved from slots 6 and 7, and since user 3 is using a (4,2)-code, its data block can be decoded as well. The same SIC process is applied, removing coded packets from user 3 to slots 4 and 5. Assume this time, that the SIC fails on slot 4: for user 2, the coded packets from slots 3 and 5 are now known, hence the data block of user 2 can be decoded, as a (3,2)-code is used.

[3]This is based on the implicit consideration that any K-MPR scheme translates to a classical transmission scheme if one user transmits K packets simultaneously: in information theoretic terms, one would not expect to increase maximum total capacity in general through this scheme (maintaining other parameters equal, e.g., the total transmission power), hence the successful transmission of K packets in parallel is expected to come at a cost of a decrease of the data rate of the packets by a factor K at least.

TABLE I
Main Parameters and Variables

General CSA Notations

Notation	Description
M	Number of users
k	Number of (uncoded) packets for each user. kN is the number of slots in one frame
$C_i(n_i, k)$	Linear block code of length n_i and rank k. The index i denotes the chosen code in a predefined set of codes
L	Maximum code length

Successive Interference Cancellation Model

Notation	Description
γ	SIC efficiency: the probability to correctly cancel the interference of one packet on one slot
w_ℓ	Probability that a packet can be correctly retrieved after SIC when ℓ packets collided in a slot

CSA Performance Metrics and Objectives

Notation	Description
G	Load of the system: $G \triangleq \frac{M}{N}$
S	Number of decoded users at the end of the iterative process. It satisfies $S \leq M$
T	Throughput of the system: $T \triangleq \frac{S}{N}$
P_L	Packet loss rate (PLR): the proportion of non-decoded users at the end of the iterative process, $P_L \triangleq 1 - \frac{S}{M}$
α	User recovery ratio
G^*_α	Load threshold: the maximum load such that a maximum ratio of $1 - \alpha$ users are non-decoded

Density Evolution Analysis

Notation	Description
Λ	User node degree distribution: $\Lambda = \{\Lambda_\ell\}_{\ell=k+1}^{L}$
ψ	Slot node degree distribution: $\psi = \{\psi_\ell\}_{\ell=0}^{M}$
λ	Polynomial representation of the user edge degree distribution: $\lambda(x) \triangleq \sum_\ell \lambda_\ell x^{\ell-1}$
ρ	Polynomial representation of the slot edge degree distribution: $\rho(x) \triangleq \sum_\ell \rho_\ell x^{\ell-1}$
R	Rate of the distribution: $R = k \sum_\ell \frac{\lambda_\ell}{\ell}$
n	Average number of packets per slot: $n \triangleq \frac{G}{R}$
p_i	Probability that a randomly chosen edge (modeling a packet) cannot be retrieved at the i-th iteration of the iterative process by applying SIC on its slot
q_i	Probability that a randomly chosen edge cannot be decoded at the i-th iteration of the iterative process by applying MAP erasure for the corresponding user
$p_{\min}$	Probability of wrongly decoding a packet at the end of the iterative process

If a data block can be successfully decoded, we remove its corresponding edges in the above bipartite graph. The decoding process consists in removing as many edges as possible from the graph to yield a maximal throughput. Density evolution is a probabilistic model of this removal process that estimates how many edges have been removed after each iteration.

B. Degree Distributions

Section II-A defines Λ_ℓ as the probability that the code (ℓ, k) is selected at random by each user. On the bipartite graph

Fig. 2, the variable $\Lambda = \{\Lambda_\ell\}_{\ell=k+1}^{L}$ can also be seen as the user node degree distribution: the amount of coded packets for a user is equal to the degree of its corresponding user node. We use $\psi = \{\psi_\ell\}_{\ell=0}^{M}$ to represent the slot node degree distribution: it is the number of packets colliding in a slot. Following [6] and the traditions of the LDPC coding literature (see [10, Eqn. (3.21)] for instance), we define the polynomial representations of these distributions which will be helpful for further calculations:

$$\Lambda(x) \triangleq \sum_{\ell=k+1}^{L} \Lambda_\ell x^\ell \qquad \text{and} \qquad \psi(x) \triangleq \sum_{\ell=0}^{M} \psi_\ell x^\ell. \quad (4)$$

Those are also known in probability theory as *probability-generating functions* of random variables Λ, and ψ. For mathematical convenience, it is also customary to define degree distributions from an edge perspective: λ_ℓ is the probability that an edge is connected to a degree-ℓ user node, while ρ_ℓ is the probability that an edge is connected to a degree-ℓ slot node. It follows from the above definitions [6] that:

$$\lambda_\ell = \frac{\Lambda_\ell \ell}{\sum_\ell \Lambda_\ell \ell} \qquad \text{and} \qquad \rho_\ell = \frac{\psi_\ell \ell}{\sum_\ell \psi_\ell \ell}. \quad (5)$$

Eqn. (5) is obtained as follows: by definition, λ_ℓ is the number of edges belonging to degree-ℓ user nodes divided by the total number of edges. The former is $\ell M \Lambda_\ell$ because there are, on average, $M\Lambda_\ell$ user nodes of degree-ℓ, and each of them has ℓ edges. The latter is simply the sum of those over all degrees ℓ, that is: $\sum_\ell \ell M \Lambda_\ell$. Writing the ratio, one obtains (5) for λ_ℓ and similarly for ρ_ℓ. Their polynomial representations are:

$$\lambda(x) \triangleq \sum_\ell \lambda_\ell x^{\ell-1} \qquad \text{and} \qquad \rho(x) \triangleq \sum_\ell \rho_\ell x^{\ell-1},$$

respectively. These definitions also verify:

$$\lambda(x) = \frac{\Lambda'(x)}{\Lambda'(1)} \qquad \text{and} \qquad \rho(x) = \frac{\psi'(x)}{\psi'(1)}, \quad (6)$$

where $\Lambda'(x) \triangleq \frac{\mathrm{d}\Lambda(x)}{\mathrm{d}x}$ is the first derivative of the function $\Lambda(x)$, and also $\psi'(x) \triangleq \frac{\mathrm{d}\psi(x)}{\mathrm{d}x}$.

Note that $R \triangleq \frac{k}{\sum_\ell \Lambda_\ell \ell} = \frac{k}{\Lambda'(1)}$ is the rate of the distribution Λ. Using (5), we see that $\sum_\ell \frac{\lambda_\ell}{\ell} = \frac{\sum_\ell \Lambda_\ell}{\sum_\ell \Lambda_\ell \ell} = \frac{1}{\Lambda'(1)}$, and we get from the edge perspective:

$$R = k \sum_\ell \frac{\lambda_\ell}{\ell}, \quad (7)$$

which shows that R is linear in λ, which is important and will be helpful for the formulation of a linear optimization problem. Note that L is the degree of the polynomial Λ, and that the index ℓ starts with the value $\ell = k+1$ corresponding to our highest-rate code $(k+1, k)$, up to the value $\ell = L$, corresponding to our lowest-rate code (L, k). In particular, for IRSA, obtained by selecting $k = 1$, the lowest index is $\ell = 2$ as in [6]. Finally, we define the average number of packets per slot as: $n \triangleq \frac{G\Lambda'(1)}{k} \triangleq \frac{G}{R}$ for further simplifications.

C. Density Evolution

Similarly to the error-free CSA resolution, let p be the probability that a randomly chosen edge (modeling a packet) cannot be retrieved yet by applying SIC on its slot. Similarly, let q be the probability that a randomly chosen edge cannot be decoded yet by applying MAP erasure for the corresponding user. At the i-th iteration of the iterative process of SIC, we denote these probabilities by p_i and q_i, respectively. Density evolution consists in studying the evolution of the probabilities on the edges, p_i and q_i, during the iterative decoding process, and their possible convergence at the end.

It was used in [8], [9] for an application of the analysis of *Low-Density Parity-Check* codes, and applied to IRSA by pioneering work in [6]. In error-free CSA, the goal is to optimize G such that $\lim_{i\to\infty} p_i = 0$, and as a result $\lim_{i\to\infty} q_i = 0$ [6]. From the definition of p, a convergence towards 0 means all packets have been decoded with probability 1. As common in the IRSA and CSA literature (see for instance [11], [13]–[17], [28]–[30]), we reuse some already known reasonings on the density evolution of p_i and q_i in similar contexts, but adapt them due to the changes in the settings, and obtain new density evolution equations such as (11).

In our model, the study of the convergence is similar but more complicated since we cannot guarantee the decoding of all users due to γ, as indicated in Section II-C. Indeed, the definition of p is skewed by the error on SIC that makes the convergence towards 0 impossible: there is a non-zero probability to wrongly retrieve each packet after SIC. We derive the two *extrinsic information transfer* (EXIT) functions [11], [31] modeling the evolution of $\{p_i\}_{i\geq 0}$ and $\{q_i\}_{i\geq 0}$, and combine them to study the monotonicity of $\{p_i\}_{i\geq 0}$ and its convergence.

To retrieve a packet by applying SIC in a slot, all the other colliding packets must have been retrieved with SIC correctly carried out. Thus, by fixing ℓ the number of packets colliding in a slot, the probability to retrieve the last packet with SIC is $1 - p_i = w_\ell (1 - q_i)^{\ell - 1}$ as shown in [6, Eqn. (8)]. In our case, with the specific w_ℓ that we defined in (1), have $1 - p_i = (\gamma(1 - q_i))^{\ell-1}$, and we can obtain new closed-form expressions: by taking the average over ℓ, $1 - p_i = \sum_\ell \rho_\ell (\gamma(1-q_i))^{\ell-1} = \rho(\gamma(1-q_i))$. The slot degree distribution $\psi(x)$ corresponds to the distribution of the number of users per slot when there are M users, each of them transmitting an average of $\frac{n}{M}$ packets in one given slot. It is the Binomial distribution $B(M, \frac{n}{M})$, hence: $\psi_\ell = \binom{M}{\ell} (\frac{n}{M})^\ell (1 - \frac{n}{M})^{M-\ell}$. Following a reasoning from the IRSA and CSA literature[4], proposed in [6]:

$$\begin{aligned}\psi(x) &= \sum_{\ell=0}^{\ell=M} \psi_\ell x^\ell = \sum_{\ell=0}^{\ell=M} \binom{M}{\ell} \left(\frac{n}{M}\right)^\ell \left(1 - \frac{n}{M}\right)^{M-\ell} x^\ell \\ &= (1 - \frac{n}{M}(1-x))^M.\end{aligned}$$

[4]It is actually an application of the classical Poisson approximation, i.e., the Poisson distribution with parameter $\mu = \alpha\beta$ can be used as an approximation of the binomial distribution $B(\alpha, \beta)$ for large α and small β. In our case, $\alpha = M$, $\beta = \frac{n}{M}$ and $\mu = n$.

When $M \to \infty$ while n is fixed:

$$\lim_{M\to\infty} \psi(x) = \lim_{M\to\infty} (1 - \frac{n}{M}(1-x))^M = e^{-n(1-x)}.$$

Coincidently, $\lim_{M\to\infty} \psi'(x) = ne^{-n(1-x)}$. Thus, we can deduce the following property for $\rho(x)$ using (6):

$$\lim_{M\to\infty} \rho(x) = \lim_{M\to\infty} \frac{\psi'(x)}{\psi'(1)} = \frac{ne^{-n(1-x)}}{ne^0} = e^{-n(1-x)}.$$

This yields the approximation: $\rho(x) \approx e^{-n(1-x)}$, where $n = \frac{G}{R}$, as in [6, Eqn. (5)]. Note that this formula becomes exact when M tends to infinity. This leads to our first EXIT equation:

$$p_i = 1 - e^{-n(1-\gamma+\gamma q_i)}. \tag{8}$$

The second EXIT equation can be found in [11, Eqn. (29)] and is obtained by studying the conditions for a packet to not be decoded by MAP erasure. For k-CSA and a fixed ℓ (degree of the user node), similar to [15], we have:

$$q_i = \sum_{j=0}^{k-1} \binom{\ell-1}{j} (1-p_{i-1})^j \, p_{i-1}^{\ell-1-j}, \tag{9}$$

where $\ell \in [k+1, L]$, since at least k packets of one user must have been decoded to perform MAP erasure. To simplify notations, we introduce the function $f_k^{(\ell)}$ such that $q_i \triangleq f_k^{(\ell)}(p_{i-1})$. By taking the average over ℓ in (9),

$$q_i = \sum_{\ell=k+1}^{L} \lambda_\ell f_k^{(\ell)}(p_{i-1}) \triangleq \hat{\lambda}(p_{i-1}). \tag{10}$$

The above function $\hat{\lambda}$ is introduced for mathematical convenience, and note that $\hat{\lambda} = \lambda$ for IRSA. We will see its usefulness later. Notice also that the EXIT function (10) is the same as in error-free CSA, whereas the EXIT function (8) is different: it takes into account the risk of erroneous SIC. We combine (8) and (10) which yields our following new recurrence relation:

$$p_i = 1 - e^{-n(1-\gamma+\gamma\hat{\lambda}(p_{i-1}))}. \tag{11}$$

The convergence condition of the sequence is: $p_{i+1} < p_i, \forall i$. However, since there is a non-zero probability of wrongly recovering every edge due to SIC error, it is impossible to have $p_i \to 0$ for $\gamma < 1$. Thus, we define an *error floor* $p_{\min}$ as the limit of the sequence $\{p_i\}_{i\geq 0}$. Note that $p_{\min}$ can be seen as the average probability of wrongly decoding a packet in a slot at the end of the process. The iterative process converges if and only if:

$$p > 1 - e^{-n(1-\gamma+\gamma\hat{\lambda}(p))}, \ \forall\, p \in \, (p_{\min}, 1]\,.$$

The convergence equation can be re-written as:

$$\hat{\lambda}(p) < -\frac{1}{\gamma}\frac{R}{G}\ln(1-p) + \frac{\gamma-1}{\gamma}, \ \forall\, p \in \, (p_{\min}, 1)\,. \tag{12}$$

The above inequality (12) is linear in λ since both R and $\hat{\lambda}(p)$ are linear in λ, according to (7) and (10). As a consequence, it can be solved using standard linear optimization solvers. Notice that when $p \to 0$, (12) will lead to $-\frac{1-\gamma}{\gamma} > 0$ which is strictly impossible for $\gamma < 1$. This confirms that for fixed G and Λ, there is an interval for p close to 0 where the inequality cannot be satisfied. This observation justifies mathematically that the limit of the sequence $\{p_i\}_{i\geq 0}$ is strictly greater than 0. It can also be noted that in the 1-MPR error-free case, the inequality is: $\hat{\lambda}(p) < -\frac{R}{G}\ln(1-p), \forall p \in (0,1)$, which is a special case of our problem with errors: the optimal load of the system has to be lower in our configuration.

An initial idea would be to fix the user recovery ratio α and find the corresponding optimal load G_α^* and code probability distribution satisfying (2) and (3). Note that α and $p_{\min}$ are both related to the PLR P_L. In this asymptotic setting, there exists a direct mapping between α and $p_{\min}$ which is:

$$1 - \alpha = P_L = \hat{\Lambda}(p_{\min}). \tag{13}$$

If we were to solve the optimization problem for a given α, we would need to reverse the function $\hat{\lambda}$, resulting in non-linear constraints. For this reason, we study the optimization problem which consists in finding the optimal $G_{p_{\min}}^*$ and Λ for a fixed and given parameter $p_{\min}$. Since $p_{\min}$ and the PLR P_L are directly related by an increasing function $\hat{\lambda}$, we show in the next section how P_L can be computed from $p_{\min}$. As a consequence, solving the aforementioned problem given $p_{\min}$ is equivalent to finding the optimal $G_{p_{\min}}^*$ and Λ for a fixed $P_L = 1 - \alpha$. In Section VI, we will confirm by simulations that this relation is also valid for finite frames.

D. Linear Programming Formulation

We derive a linear optimization problem from the convergence equation (12), which is formulated as follows:

$$\begin{aligned}
\underset{\lambda_i}{\text{Maximize}} \quad & G & (\mathcal{P}_1)\\
\text{subject to:} \quad & C1 : \forall i \in [k+1, ..., L], \ 0 \leq \lambda_i \leq 1,\\
& C2 : \sum_{i=k+1}^{L} \lambda_i = 1,\\
& C3 : G\hat{\lambda}(p) < -\frac{1}{\gamma}R\ln(1-p) + G\frac{\gamma-1}{\gamma},\\
& \qquad \forall\, p \in \, (p_{\min}, 1)\,.
\end{aligned}$$

The objective is to maximize the system load G. Constraints $C1$ and $C2$ are mandatory to make sure λ is a probability distribution, while $C3$ comes from (12). To limit ourselves to a finite number of inequalities, we use a technique introduced for low-density parity-check codes (LDPC) [32] which consists in taking a finite set of values of p, by discretizing the interval $(p_{\min}, 1)$ into steps of size s.

Since R is linear in λ, the feasibility problem (i.e., checking whether $C1$, $C2$ and $C3$ are feasible) is linear in λ. Increasing G leads to an increase in the left-hand term of $C3$, and there is a maximum load threshold $G_{p_{\min}}^*$ such that the problem is not feasible for $G > G_{p_{\min}}^*$. We proceed by bisection on G: for each value of G, we check whether the problem is feasible and increase or decrease G accordingly.

We present the pseudo-code of this procedure in Algorithm 1. The algorithm takes as input ϵ, the error tolerance at termination of the bisection search. The initial values $G_{\min} \leftarrow 0$ and $G_{\max} \leftarrow 1$ are chosen so that $G_{\min}$ is feasible

and $G_{\max}$ is a theoretical upper-bound of $G^*_{p_{\min}}$. The bisection search stops whenever a solution ϵ-close to the optimal is found. This only requires $O(\log_2(1/\epsilon))$ iterations.

Algorithm 1 Load maximization bisection search

input: ϵ
initialization: $G_{\min} \leftarrow 0$, $G_{\max} \leftarrow 1$
while $G_{\max} - G_{\min} > \epsilon$ **do**
  $G \leftarrow (G_{\min} + G_{\max})/2$
  Run a linear solver to check the feasibility of $\mathcal{P}_1$
  given a load G
  if $\mathcal{P}_1$ is feasible **then**
    $G_{\min} \leftarrow G$
  else
    $G_{\max} \leftarrow G$
  end if
end while
return $G_{\min}$

Through this procedure, the objective converges to the value of the maximum threshold $G^*_{p_{\min}}$. The solver returns an optimal λ which is then mapped to the corresponding code probability distribution Λ thanks to (5). It is also possible to obtain a similar linear problem for 2-MPR. The details of the 2-MPR case are presented in Section IV.

If one is interested in working at a fixed rate R_{fixed}, the problem formulation $\mathcal{P}_1$ can be extended by adding a new linear constraint $R = R_{\text{fixed}}$.

A common approach found in the literature to solve load maximization problems is to use the differential evolution heuristic. However, this method is not guaranteed to find the optimal distributions. Indeed, we see that [16, Table I] contains distributions (e.g., for $K = 1$ and degree $n = 8$ and $n = 16$) outperforming the ones found by differential evolution in [6, Table I]. Another advantage of our linear solver is that the precision of the solution can be controlled by the number of bisection steps and the discretization granularity s.

IV. Density Evolution for 2-MPR

Here, we extend the results developed in the previous section to the 2-MPR case. We show in Section IV-A that the load maximization problem is overall linear except for one non-linear constraint. In Section IV-B, we propose a sharp linear approximation of the non-linear constraint, which enables to solve the 2-MPR problem using the bisection search (Algorithm 1), similarly to the 1-MPR case.

A. Optimization Problem Formulation

The load maximization problem for 2-MPR is formulated below. It is similar to the problem formulation $\mathcal{P}_1$ for 1-MPR. The constraints $C1$ and $C2$ are the same. Constraints $C3'$ and $C4$ are different from those for the 1-MPR case. W_{-1} denotes the second real-branch of the Lambert function [33]. The derivations of these new constraints are given in Appendix A.

$$\begin{aligned}
&\underset{\lambda_i}{\text{Maximize}} \quad G && (\mathcal{P}_2)\\
&\text{subject to:} \quad C1: \forall i \in [k+1, ..., L],\ 0 \le \lambda_i \le 1,\\
&\qquad C2: \sum_{i=k+1}^{L} \lambda_i = 1,\\
&\qquad C3': \forall p \in (p_{\min}, ..., 1),\\
&\qquad\quad \hat{\lambda}(p) < -\frac{1}{n\gamma} W_{-1}\left(\frac{\gamma(p-1)}{e^{\gamma}} e^{n-n\gamma}\right) - \frac{1}{n},\\
&\qquad C4: G < R\left(-\frac{\ln(1-p_{\min})}{1-\gamma}\right).
\end{aligned}$$

We can see that $C3'$ is non-linear, while the rest of the problem is linear, including $C4$. To solve efficiently this problem, we propose in the next subsection a linear approximation of constraint $C3'$.

B. Near-Optimal Linear Approximation

We will derive a linear upper-bound of W_{-1}. Once such an upper-bound is found, it can replace W_{-1} in constraint $C3'$ to make problem $\mathcal{P}_2$ linear.

We have the following inequality for all p and n:

$$W_{-1}\left(\frac{\gamma(p-1)}{e^{\gamma}} e^{n-n\gamma}\right) < \ln(\gamma(1-p)) + n(1-\gamma) - \gamma.$$

This inequality comes from a linear bound of W_{-1} found by Faris Alzahrani & Ahmed Salem in [34]:

$$\forall t > 0,\ W_{-1}(-e^{-(t+1)}) < -(t+1), \tag{14}$$

where

$$-(t+1) \triangleq \ln(\gamma(1-p)) + n(1-\gamma) - \gamma. \tag{15}$$

The proof of (14) uses the fact that function $t \mapsto W_{-1}\left(-e^{-(t+1)}\right)$ is convex. Therefore, it can be upper bounded between any two points by the straight line crossing these two points. The two points considered to obtain (14) in [34] are $t_{\min} = 0$ and $t_{\max} = \infty$. If we restrict the values of t to smaller intervals of the form $[t_{\min}, t_{\max}]$, then the inequality can be made even sharper:

$$\forall t \in [t_{\min}, t_{\max}],\ W_{-1}\left(-e^{-(t+1)}\right) \le -\alpha\, t - \beta, \tag{16}$$

where

$$\alpha = \frac{W_{-1}\left(-e^{-(t_{\min}+1)}\right) - W_{-1}\left(-e^{-(t_{\max}+1)}\right)}{t_{\max} - t_{\min}}, \tag{17}$$

$$\beta = -\alpha\, t_{\min} - W_{-1}\left(-e^{-(t_{\min}+1)}\right). \tag{18}$$

In our case, t is obtained by a change of variables and depends on p and n, see (15). For $p \in (p_{\min}, ..., 1)$, t belongs to $[t_{\min}(p), t_{\max}(p)]$ such that:

$$t_{\min}(p) = -\ln(\gamma(1-p)) - (n_{\max}+1)(1-\gamma), \tag{19}$$

$$t_{\max}(p) = -\ln(\gamma(1-p)) - (1-\gamma), \tag{20}$$

$$n_{\max} = -\frac{\ln(1-p_{\min})}{1-\gamma}. \tag{21}$$

Since α, β are functions of $t_{\min}$ and $t_{\max}$, we denote them by α_p and β_p for simplicity. We derive the following inequalities for constraints $C3'$:

$$\begin{aligned}
\hat{\lambda}(p) &< -\frac{1}{n\gamma} W_{-1}\left(\frac{\gamma(p-1)}{e^{\gamma}} e^{n-n\gamma}\right) - \frac{1}{n},\\
&< -\frac{(-\alpha_p\, t - \beta_p)}{n\gamma} - \frac{1}{n},
\end{aligned} \tag{22}$$

$$< -\frac{(-\alpha_p(-\ln(\gamma(1-p)) - (n+1)(1-\gamma)) - \beta_p)}{n\gamma} - \frac{1}{n}, \tag{23}$$

$$< -\frac{\alpha_p}{n\gamma}\ln(\gamma(1-p)) + \frac{\alpha_p(\gamma-1)}{\gamma} + \frac{\alpha_p(\gamma-1)+\beta_p-\gamma}{n\gamma}.$$

We derive inequality (22) from (18). Inequality (23) is obtained be replacing t with its definition in (15).

Using $n \triangleq G/R$, we get the following sharp piece-wise linear upper-bound of $C3'$:

$$\forall p \in (p_{\min},\ldots,1), \quad \hat{\lambda(p)} < -\frac{\alpha_p}{\gamma} R \ln(\gamma(1-p)) + \alpha_p G \frac{\gamma-1}{\gamma} + R\left(\frac{\alpha_p(\gamma-1)+\beta_p-\gamma}{\gamma}\right).$$

Appendix B provides a plot (Fig. 14) that can show how close the function and its approximation are. By replacing $C3'$ with its linear upper-bound, we obtain the following linear optimization problem, which is a close approximation of $\mathcal{P}_2$.

$$\underset{\lambda_i}{\text{Maximize}} \quad G \qquad (\mathcal{P}_2')$$

$$\text{subject to:} \quad C1: \forall i \in [k+1,...,L],\ 0 \le \lambda_i \le 1,$$

$$C2: \sum_{i=k+1}^{L} \lambda_i = 1,$$

$$C3': \forall p \in (p_{\min},...,1),$$

$$G\hat{\lambda(p)} < -\frac{\alpha_p}{\gamma} R \ln(\gamma(1-p)) + \alpha_p G \frac{\gamma-1}{\gamma} + R\left(\frac{\alpha_p(\gamma-1)+\beta_p-\gamma}{\gamma}\right),$$

$$C4: G < R\left(-\frac{\ln(1-p_{\min})}{1-\gamma}\right).$$

Checking the feasibility of problem $\mathcal{P}_2'$ can now be done by linear programming. Hence, we solve it using the bisection search presented in Algorithm 1. This allows us to optimize the design of CSA for 2-MPR.

V. Design of Coded Slotted ALOHA

We apply Python CVXPY linear solver [35] on the optimization problem ($\mathcal{P}_1$), and study the impact of each parameter to optimize the design of CSA, starting with 1-MPR. The error tolerance at termination of the bisection search ϵ is set to 0.001 while the discretization step s is set to 0.02.

The SIC efficiency γ is a parameter imposed by the system. We explore the impact of its influence in Fig. 3. Because we are systematically studying the impact of each parameter, we use one fixed value for γ for other figures. We selected the value $\gamma = 0.99$: it is approximately one order of magnitude higher than the mentioned results in [6, Appendix B] and [11, Appendix A], whose physical layer simulations report a fairly low decoding error rate (10^{-2} and 10^{-3} after 7 SIC operations). It corresponds to one use case of CSA for high-reliability communications (as in URLLC [36]). Considering only the impact of the imperfect CSI, from [22] for instance, we know that the reliability of the SIC process can be improved and approach the performance of perfect CSI, by using a dedicated pilots-based channel estimation with OFDM at the expense of having to operate at higher SNR.

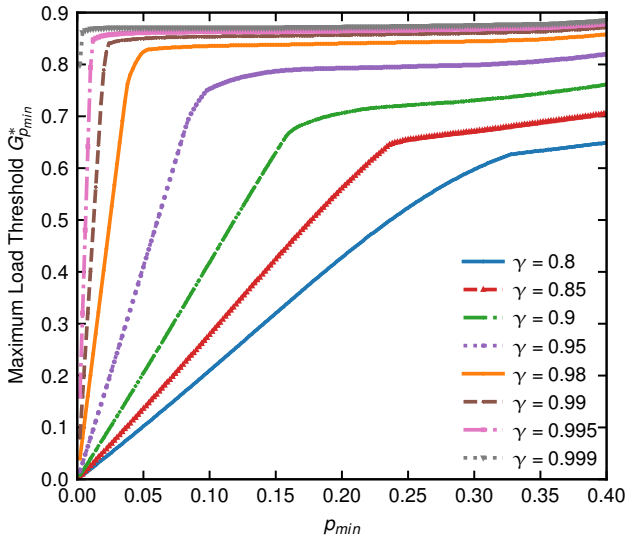

Fig. 3. Influence of parameter γ (1-MPR): Maximum load threshold $G^*_{p_{\min}}$ versus $p_{\min}$ for 3-CSA, $L = 15$, $R = 0.4$ and different γ.

The other parameters to study and optimize are the rate R, the maximum code length L, the CSA parameter k, and the error floor $p_{\min}$ which is related to the PLR P_L by (13). In Fig. 3 to Fig. 9, we plot the maximum load threshold G^* as a function of the parameter $p_{\min}$ for different values of the parameter under study, and fix all the other parameters to preset values. Due to the relation between $p_{\min}$ and P_L, the x-axis representing $p_{\min}$ in these figures can be seen as an increasing function of the packet loss rate P_L.

Before fixing $\gamma = 0.99$, we start by studying the influence of this SIC efficiency on the system performance. In Fig. 3, we set $k = 3$, $L = 15$, impose the rate $R = 0.4$ and plot the maximum load threshold $G^*_{p_{\min}}$ as a function of $p_{\min}$. The curves have similar shapes, an increase in $p_{\min}$ leads to an increase in the maximum load $G^*_{p_{\min}}$ since the problem is less constrained. The curves also all have an inflection point, the load increases significantly below this point and saturates around 0.9 after. If we decrease γ, the maximal load decreases and is reached for a higher $p_{\min}$, i.e., a higher packet loss rate. We would also get similar shapes of curves for other values of k. Observing the similarity of the shape of the curves, in the rest of the study, we select for γ the value $\gamma = 0.99$. The study of CSA optimization can easily be generalized to other values of γ, and the observed phenomena would be the same.

With the same parameters, we study the influence of the rate R in Fig. 4. A known result from [11] is that CSA heavily outperforms IRSA for $R > \frac{1}{3}$, and it is even impossible to reach $R > \frac{1}{2}$ with IRSA. Moreover, G and R are inversely proportional, and a high load is often offset by a poor performance in terms of rate, which means poor energy performance. However, our results show that the best performance is achieved for $R = 0.4$, which confirms the benefits of 3-CSA. $R = 0.25$ would provide better results only by increasing the maximum code length L, which would lead to a lower local rate $\frac{k}{L}$. It is also worth noting that at low load, distributions with higher rate perform better in terms of PLR.

Another parameter to study is the maximum code length

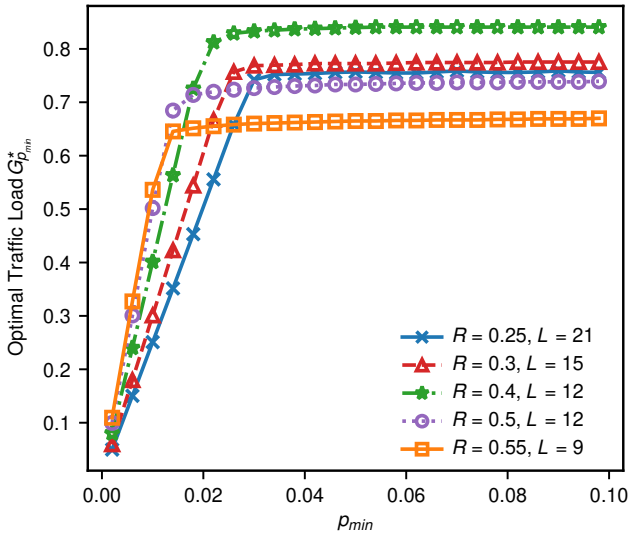

Fig. 4. Influence of parameter R: Maximum load threshold $G^*_{p_{\min}}$ versus $p_{\min}$ for 3-CSA, $\gamma = 0.99$ and different R.

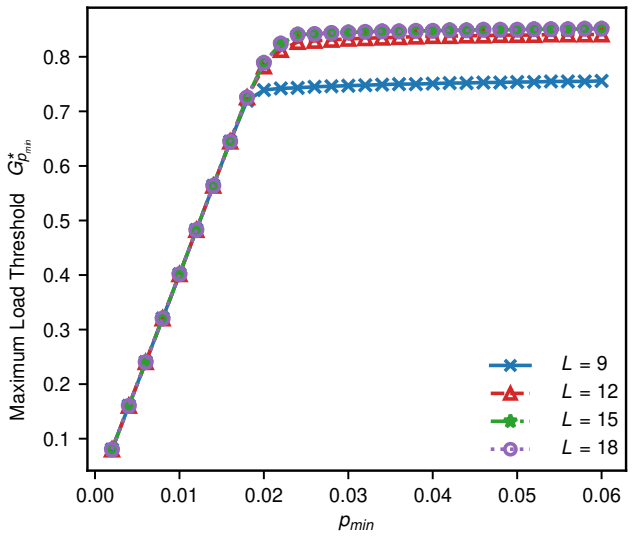

Fig. 5. Influence of parameter L: Maximum load threshold $G^*_{p_{\min}}$ versus $p_{\min}$ for 3-CSA, $\gamma = 0.99$, $R = 0.4$ and different L.

L. It is logical that increasing L leads to higher performance for the load of the system, but at the cost of a lower local rate, leading to higher energy consumption. It can be seen from Fig. 5 obtained with 3-CSA that increasing L from 9 to 12 (local rate from $\frac{1}{3}$ to $\frac{1}{4}$ since $k = 3$) allows the load to increase from 0.75 to 0.85 but it saturates after $L = 15$. Overall, a minimum local rate of $\frac{1}{5}$ seems to be a fair choice.

We now look at the influence of CSA parameter k, on the load but also on the rate of the corresponding optimal distribution Λ. The state-of-the-art on CSA shows that a higher k allows to reach higher loads (closer to 1) for high-rates distributions. It can be seen from Fig. 6 that for an imposed rate $R = 0.4$, CSA performs better with a high number of encoded packets k. As expected, IRSA ($k = 1$) performs very poorly at this rate. In Fig. 7, we do not impose the rate and see if better distributions can be found for lower rates. Indeed we see that k-CSA with lower k performs better, but at the expense of having a low rate. The conclusion is that a low k

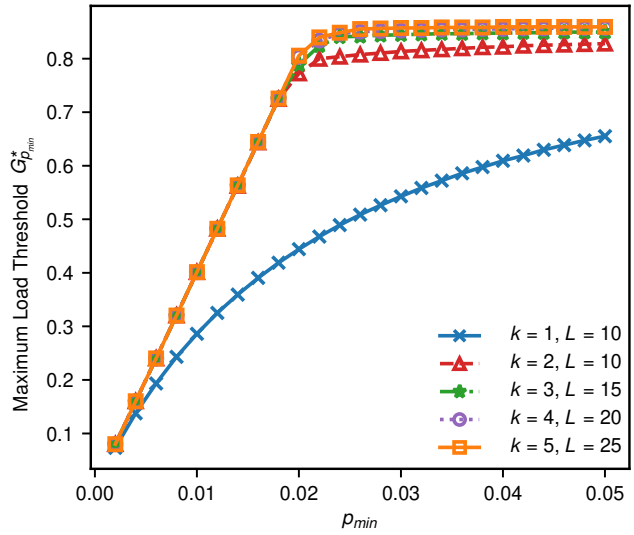

Fig. 6. Influence of parameter k: Maximum load threshold $G^*_{p_{\min}}$ versus $p_{\min}$ for $\gamma = 0.99$, $R = 0.4$ and different k.

leads to the best load performance by employing a low rate distribution, but a high k is more suited if the rate is fixed and relatively high.

For error-free CSA, the parameter $p_{\min}$ is nonexistent, and the problem returns a distribution that leads to the optimal load for an asymptotic zero-PLR. For error-free IRSA, the (truncated) Soliton distributions have been shown to reach the optimal $G^* \to 1$ packet/slot asymptotically [7], while for error-free k-CSA, optimized distributions have been found by the differential evolution metaheuristic and proposed in [11]. Since we are considering CSA under a new set of assumptions, it is of prime interest to compare the performance with that of such distributions obtained under error-free assumptions, denoted *optimal error-free distributions*. We indeed will show that they are no longer "optimal" when the error-free assumption is actually not realized. The results are shown in Table II: for each $p_{\min}$, we solve the optimization problem and obtain a distribution that we denote by $\Lambda_{p_{\min}}$. This distribution corresponds to a maximum load threshold $G^*_{p_{\min}}$, a rate R and an asymptotic PLR P_L given by (13). The optimal error-free distribution is also obtained with the inputs $\gamma = 1$ and $p_{\min} = 0$: the results are referenced in the bottom line. A first result is that when increasing $p_{\min}$, the load G increases and the corresponding distributions Λ also vary depending on $p_{\min}$. For $p_{\min} \leq 0.025$, the optimal distribution is a monomial which is analogous to a known result on LDPC codes [32]. On the other hand, when $p_{\min}$ gets larger, the distributions seem to converge towards the optimal error-free distribution, and the rate remains the same. If the optimal error-free distribution is reached by significantly increasing $p_{\min}$, this suggests it provides poor performance with regards to $p_{\min}$, and thus in terms of the PLR P_L. This observation will have to be confirmed in Section VI. The asymptotic packet loss rates are shown in the right column. Although these values are low, we can still see variations of factor 10 to 100 that will be much more significant when considering finite ALOHA. The same study is conducted for k-CSA with similar results: distributions vary depending on $p_{\min}$ and converge towards

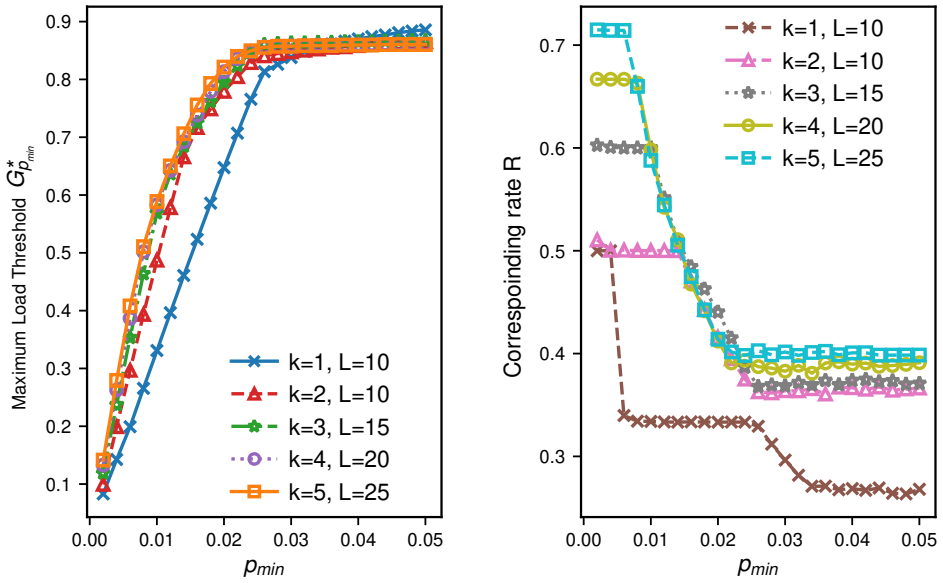

Fig. 7. Influence of parameter k: Maximum load threshold $G^*_{p_{\min}}$ and corresponding rate R versus $p_{\min}$ for $\gamma = 0.99$ and different k.

$p_{\min}$	R	$\Lambda(x)$	$G^*_{p_{\min}}$	PLR P_L
0.005	0.5	x^2	0.167	2.5×10^{-5}
0.01	0.33	$0.01x^2 + 0.99x^3$	0.331	2.0×10^{-6}
0.015	0.33	x^3	0.492	3.4×10^{-6}
0.02	0.33	x^3	0.647	8.0×10^{-6}
0.025	0.33	x^3	0.794	1.6×10^{-5}
0.03	0.3	$0.94x^3 + 0.01x^9 + 0.05x^{10}$	0.838	2.5×10^{-5}
0.035	0.27	$0.01x^2 + 0.88x^3 + 0.11x^{10}$	0.859	5.0×10^{-5}
0.04	0.27	$0.07x^2 + 0.81x^3 + 0.12x^{10}$	0.870	1.6×10^{-4}
0.045	0.27	$0.12x^2 + 0.76x^3 + 0.12x^{10}$	0.879	3.1×10^{-4}
0.05	0.27	$0.15x^2 + 0.72x^3 + 0.13x^{10}$	0.886	4.6×10^{-4}
0.06	0.27	$0.21x^2 + 0.66x^3 + 0.13x^{10}$	0.896	9.0×10^{-4}
0.07	0.27	$0.24x^2 + 0.62x^3 + 0.14x^{10}$	0.904	1.3×10^{-3}
0.08	0.27	$0.27x^2 + 0.58x^3 + 0.15x^{10}$	0.910	2.0×10^{-3}
0.09	0.27	$0.29x^2 + 0.56x^3 + 0.15x^{10}$	0.914	2.8×10^{-3}
0.1	0.27	$0.31x^2 + 0.54x^3 + 0.15x^{10}$	0.918	3.6×10^{-3}
0.2	0.26	$0.38x^2 + 0.43x^3 + 0.02x^4 + 0.17x^{10}$	0.930	1.9×10^{-2}
...	...	...	...	
optimal error-free	0.27	$0.52x^2 + 0.17x^3 + 0.15x^4 + 0.16x^{10}$	0.952	

TABLE II
CSA DISTRIBUTIONS FOR $k = 1$, $\gamma = 0.99$, $L = 10$

the optimal distribution in the error-free case. The tables for k-CSA are omitted here due to redundancy. Besides, additional results that compare the behavior of different distributions are shown in Appendix C.

Finally, we study the system in the 2-MPR configuration, considering the optimization problem $(\mathcal{P}'_2)$ derived. When studying the influence of γ in Fig. 8, we see that the shapes are similar but the initial slopes are approximately twice lower when compared to those of 1-MPR in Fig. 3: hence optimal loads are reached for roughly twice higher values of $p_{\min}$ which is a worse performance. Similar to that for 1-MPR, we also study the influence of k over the load threshold and the rate of the distribution in Fig. 9. Overall, our conclusions are that the results for 2-MPR and 1-MPR are similar, but the PLRs P_L for 2-MPR are much higher as the loads are reached for higher $p_{\min}$. For that reason, 1-MPR should be preferred. However, 2-MPR presents one advantage, that is to allow for distributions with higher rates than 1-MPR: for instance with IRSA the optimal rate is 0.38 for 2-MPR instead of 0.26 for 1-MPR. This rate of 0.38 is even higher than the known IRSA threshold $R \leq \frac{1}{3}$ for 1-MPR. The higher $p_{\min}$ for 2-MPR could have been foreseen, since 2-MPR involves loads close to 2 instead of 1 for 1-MPR. We attempt to double the amount of users in the system while keeping the same number of slots. This means SIC will be much more used and the probability of IC error will be higher, leading to higher PLRs P_L.

VI. Simulation Results

In the previous section, the optimization problem helps us to find various distributions $\Lambda_{p_{\min}}$ that provide different trade-offs between load and error. The results are obtained in the asymptotic configuration, when the number of users and slots tends to infinity. Now we want to confirm our results in a finite setting, by computer simulations. Although the PLRs

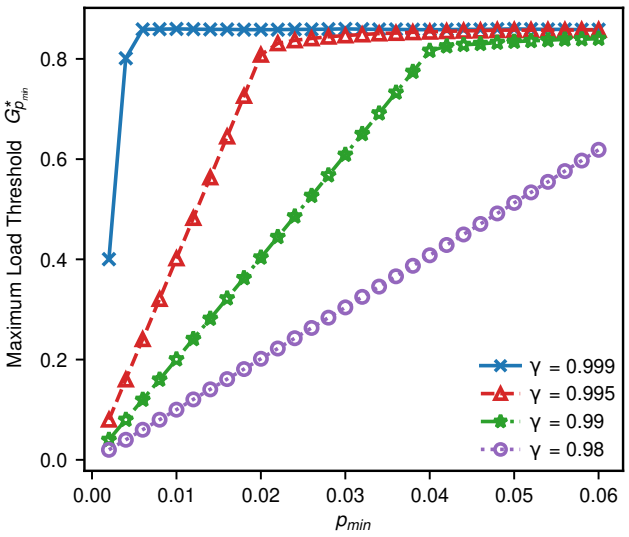

Fig. 8. Influence of parameter γ (2-MPR): Maximum load threshold G and corresponding rate R versus $p_{\min}$ for 3-CSA, 2-MPR, $R = 0.4$, $L = 15$ and different γ.

found analytically are low, we observe orders of magnitude differences between them and we will see that they are much more important and cannot be neglected when N is finite. For the following observations, we fix the amount of slots kN with $N = 1000$ and study the PLR P_L as a function of the load of the system, averaged over 1000 simulations and set $\gamma = 0.99$.

Fig. 10 is the plot of the PLR P_L as a function of the load in the case of IRSA for several distributions: $\Lambda_{0.02}$, $\Lambda_{0.05}$ and $\Lambda_{0.1}$ for $p_{\min} = 0.02$, 0.05 and 0.1, respectively, also the optimal error-free distribution. The distributions Λ with high $p_{\min}$ perform better above a certain load threshold, corresponding to the optimal load of the distribution. For instance, the load threshold of $\Lambda_{0.1}$ is 0.85: it is the load where this distribution performs better than the others. However, when the load is below the threshold, the distributions for a lower $p_{\min}$ achieve a lower PLR P_L and have better performance. This confirms that the distributions are optimal only for a specific load: other distributions may be preferable when the load of the system varies. The optimal loads are also lower in the finite setting: around 0.85 for a tolerated PLR P_L of 10^{-2} instead of 0.9. Finally, we can see that the optimal error-free distribution performs poorly: it is the best distribution for $G > 0.9$ but the PLR P_L is already above 10^{-1}. For a load below 0.9, the distributions $\Lambda_{p_{\min}}$ perform better. The relation between $p_{\min}$ and the PLR P_L is also confirmed by this figure.

To study the impact of the IC error, we also run simulations without considering this error, i.e., $\gamma = 1$. The results are presented in Fig. 11. The shapes of the curves are similar: each distribution $\Lambda_{p_{\min}}$ performs the best around the same load. This highlights that the packet loss rate is not only due to the consideration of SIC error. However, the PLRs P_L are much less pronounced when not accounting for SIC error: around 5 to 10 times lower than those in our error case ($\gamma = 0.99$). This means that taking SIC error into account makes the PLR not negligible anymore, and therefore the choice of the optimal distribution with regards to the PLR is a critical question. Besides, additional simulation results that assess the impact of frame size are shown in Appendix D.

The curves presented so far are obtained for IRSA, nevertheless we make a similar study for k-CSA, once again considering the $\Lambda_{p_{\min}}$ distributions. Fig. 12 highlights similar results in the case of 3-CSA, with distributions performing best at their optimal load, and the optimal error-free distribution performing poorly overall. For example, $\Lambda_{0.02}$ has better performance up to around $G \leq 0.7$, $\Lambda_{0.05}$ is preferable for approximately $0.75 \leq G \leq 0.85$, and the "optimal error-free" distribution dominates for $G > 0.85$. We also see that the PLRs involved here are much higher than those for IRSA shown in Fig. 10.

Finally, we compare the PLR performance of different distributions for fixed loads. For different loads $G = 0.8$, 0.825, 0.85 and 0.875, we plot in Fig. 13 the PLR P_L obtained for the different distributions $\Lambda_{p_{\min}}$, where $p_{\min} \in [0.01, 0.3]$ is represented on the x-axis. The optimal error-free distribution is also studied and plotted on the right. For a fixed load G, we can see that the minimum PLR P_L is reached for a single distribution $\Lambda_{p^*_{\min}(G)}$. For instance for $G = 0.825$, the minimum PLR P_L is reached for the distribution $\Lambda_{0.05}$, i.e., $p^*_{\min}(0.825) = 0.05$. The explanation for this unique optimum is as follows. Let us fix the load of the system G. A distribution $\Lambda_{p_{\min}}$ with $p_{\min} < p^*_{\min}(G)$ has a load threshold lower than G. The load threshold of the distribution is therefore exceeded and the PLR dramatically increases. On the other hand, for a distribution $\Lambda_{p_{\min}}$ with $p_{\min} > p^*_{\min}(G)$, the distribution performs below its load threshold so the choice is not optimal and results to slightly higher PLRs. It should be noted that when G increases, the curves are flattened and the PLR performance becomes less sensible to the choice of distributions. The key result here is that for each load G, there is a single distribution $\Lambda_{p^*_{\min}(G)}$ that achieves the lowest PLR. The optimal error-free distribution can always be outclassed by a distribution $\Lambda_{p_{\min}}$.

VII. Conclusion

We study a novel extension of coded slotted ALOHA, taking into account realistic effects, namely errors due to an imperfect SIC process. With these new assumptions and new modeling, we propose a method to obtain the optimal distributions (in terms of maximum load threshold, or equivalently, throughput) for given values of parameters such as the probability of SIC error. This is accomplished by first deriving new density evolution equations, then formulating an optimization problem, that can be solved efficiently through different techniques that we propose, including recourse to underlying linear optimization problems. Then, starting from a preset system configuration, through numerically obtained distributions, we systematically study the influence of each parameter on the performance. We consider a fixed SIC efficiency $\gamma = 0.99$, nevertheless we show that the study can be generalized to any γ. We show that increasing the dimension k of the CSA codes allows finding distributions with higher rates, although with slightly lower maximal loads. We introduce parameter $p_{\min}$, which allows

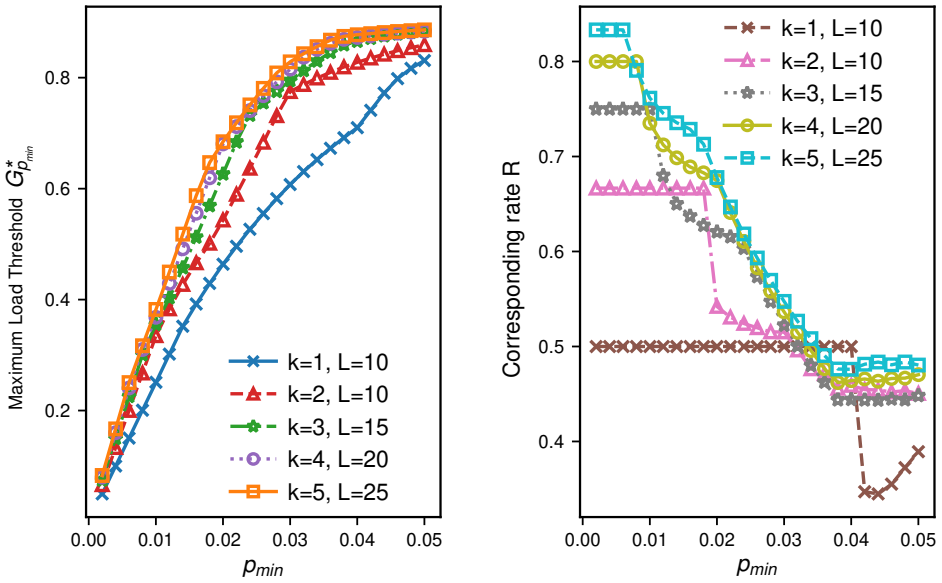

Fig. 9. Influence of parameter k (2-MPR): Maximum load threshold $G^*_{p_{\min}}$ and corresponding rate R versus $p_{\min}$ for 2-MPR, $\gamma = 0.99$ and different k.

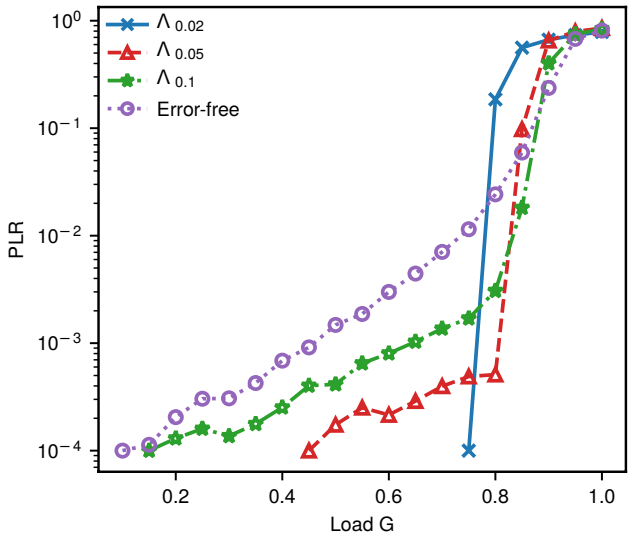

Fig. 10. PLR P_L versus load G for IRSA, $\gamma = 0.99$, $N = 1000$ and different distributions Λ.

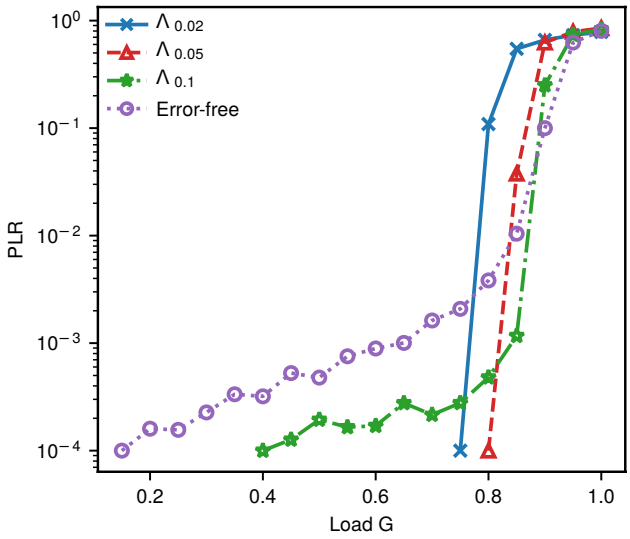

Fig. 11. PLR P_L versus load G for IRSA, $\gamma = 1$, $N = 1000$ and different distributions Λ.

us to extract new distributions $\Lambda_{p_{\min}}$, that are each optimal for a certain load G and a packet loss rate P_L. Choosing a distribution with a high load threshold is at the expense of having a high PLR as well. These PLRs are dramatically increased if SIC error is taken into account. Although the PLRs were theoretically low in the asymptotic setting, we show that they are not negligible in a finite setting. We also confirm that for a given load threshold G, there is a single distribution $\Lambda_{p^*_{\min}(G)}$ achieving the lowest packet loss rate. Therefore, this distribution should be chosen in practice if the load can be estimated. On the other hand, while the optimal error-free distribution computed without assuming SIC errors has excellent performance when sticking to the assumptions, it is quickly outclassed by $\Lambda_{p_{\min}}$ distributions as soon as a small SIC error probability is introduced and experiences poor performance, especially when applied to a non-asymptotic setting, e.g., a finite number of users and slots. Notice that we do not consider and use capture effect in our study here, which can be considered in a future work.

ACKNOWLEDGMENT

The authors would like to thank Prof. Jinhong Yuan and Zhuo Sun (UNSW Sydney) for their help and some discussions during this work. Besides, a part of the work was carried out at LINCS (www.lincs.fr).

APPENDIX

A. *Density Evolution for 2-MPR*

In this appendix, we determine the convergence equations for 2-MPR. These equations form constraints $C3'$ and $C4$ of the load maximization problem $\mathcal{P}_2$ in Sub-section IV-A.

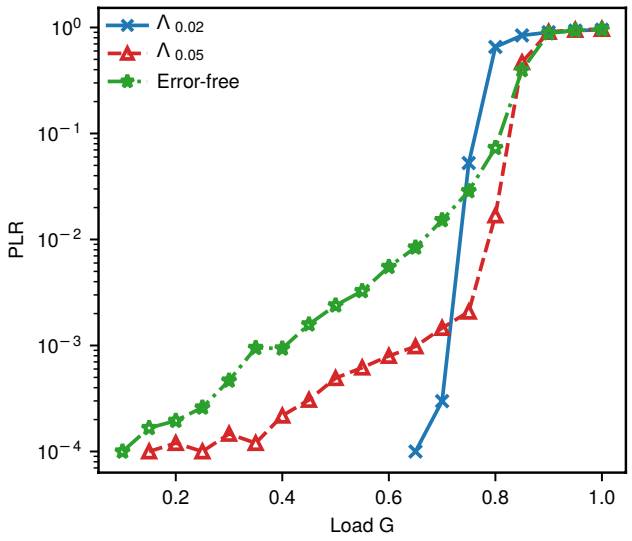

Fig. 12. PLR P_L versus load G for 3-CSA, $\gamma = 0.99$, $N = 1000$ and different distributions Λ.

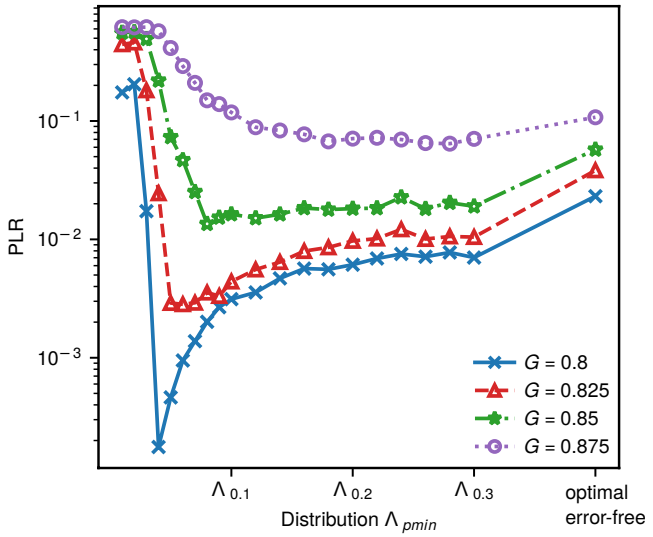

Fig. 13. PLR P_L versus $\Lambda_{p_{\min}}$ for IRSA, $\gamma = 0.99$, $N = 1000$ and different loads G.

We first start by deriving the two EXIT equations. Following the analysis of [6, Eqn. (11)], [13, Eqn. (8)] and [15, Eqn. (8)], in the 2-MPR case, two packets can be simultaneously decoded in a slot, i.e., $w_{\ell-2} = 1$. It follows that for a fixed ℓ, the probability to retrieve packets in a slot with 2-MPR is:

$$1 - p_i = w_\ell (1-q_i)^{\ell-1} + w_{\ell-1} \binom{\ell-1}{1} (1-q_i)^{\ell-2} q_i.$$

The fact that $w_\ell = \gamma^{\ell-1}$ in (1) implies:

$$1 - p_i = (\gamma(1-q_i))^{\ell-1} + (\ell-1)(\gamma(1-q_i))^{\ell-2} q_i. \quad (24)$$

By taking the average over ℓ, we get:

$$1 - p_i = \sum_\ell \rho_\ell (\gamma(1-q_i))^{\ell-1} + q_i \sum_\ell \rho_\ell (\ell-1)(\gamma(1-q_i))^{\ell-2}.$$

Since $\rho(x) = \sum_\ell \rho_\ell x^{\ell-1}$ and $\rho'(x) = \sum_\ell \rho_\ell (\ell-1) x^{\ell-2}$, we deduce that:

$$1 - p_i = \rho(\gamma(1-q_i)) + q_i \rho'(\gamma(1-q_i)).$$

When $M \to \infty$, the slot degree distribution ψ converges to a Poisson distribution [6, Eqn. (5)]. As a consequence, we can write the edge distribution approximation as: $\rho(x) = e^{-n(1-x)}$ and $\rho'(x) = n e^{-n(1-x)}$. The first EXIT equation for 2-MPR is then:

$$p_i = 1 - e^{-n(1-\gamma(1-q_i))} - q_i n e^{-n(1-\gamma(1-q_i))}.$$

The second EXIT equation is the same as for 1-MPR:

$$q_i = \hat{\lambda}(p_{i-1}).$$

Combining the two EXIT equations gives us the following recurrence relation for 2-MPR:

$$p_i = 1 - e^{-n(1-\gamma+\gamma\hat{\lambda}(p_{i-1}))}(1 + n\hat{\lambda}(p_{i-1})). \quad (25)$$

Recall that in problem $\mathcal{P}_2$, we are only interested in finding code distributions for which the recurrence relation (25) converges. The iterative process converges if and only if the following inequalities hold for all $p \in (p_{\min}, 1]$:

$$\begin{aligned} & p > 1 - e^{-n(1-\gamma+\gamma\hat{\lambda}(p))}(1 + n\hat{\lambda}(p)) \\ \iff & \gamma(p-1) > e^{-n+n\gamma-n\gamma\hat{\lambda}(p)}(-\gamma - n\gamma\hat{\lambda}(p)), \\ \iff & \gamma(p-1)\frac{e^{n-n\gamma}}{e^\gamma} > e^{-\gamma-n\gamma\hat{\lambda}(p))}(-\gamma - n\gamma\hat{\lambda}(p)). \quad (26) \end{aligned}$$

Let W_{-1} and W_0 be the real-branches of the Lambert functions [33]. By definition, W_{-1} and W_0 are the inverse functions of $x \mapsto xe^x$ for respectively $x < 1$ and $x \geq 1$. We can see that (26) is an inequality of the form $Xe^X < M < 0$, where $M = \frac{\gamma(p-1)}{e^\gamma} e^{n-n\gamma}$ and $X = -\gamma - n\gamma\hat{\lambda}(p)$. Therefore, (26) is equivalent to two inequalities:

$$W_{-1}\left(\frac{\gamma(p-1)}{e^\gamma} e^{n-n\gamma}\right) < -\gamma - n\gamma\hat{\lambda}(p), \quad (27)$$

$$W_0\left(\frac{\gamma(p-1)}{e^\gamma} e^{n-n\gamma}\right) > -\gamma - n\gamma\hat{\lambda}(p). \quad (28)$$

Constraint $C3'$ becomes apparent after isolating $\hat{\lambda}(p)$ in inequality (27):

$$\hat{\lambda}(p) < -\frac{1}{n\gamma} W_{-1}\left(\frac{\gamma(p-1)}{e^\gamma} e^{n-n\gamma}\right) - \frac{1}{n}.$$

The left hand side of (28) is increasing in p, while its right hand side is decreasing in p. Thus, satisfying (28) for all $p \in (p_{\min}, 1]$ is equivalent to just satisfy it for the border case $p = p_{\min}$:

$$W_0\left(\frac{\gamma(p_{\min}-1)}{e^\gamma} e^{n-n\gamma}\right) > -\gamma - n\gamma\hat{\lambda}(p_{\min}). \quad (29)$$

The following inequality (30) is a sufficient condition to satisfy (29). We derive it from the fact that $\hat{\lambda}(p_{\min}) \geq 0$. It is also a good approximation in practice when $p_{\min} \approx 0$, and thus $\hat{\lambda}(p_{\min}) \approx 0$.

$$W_0\left(\frac{\gamma(p_{\min}-1)}{e^\gamma} e^{n-n\gamma}\right) > -\gamma \quad (30)$$

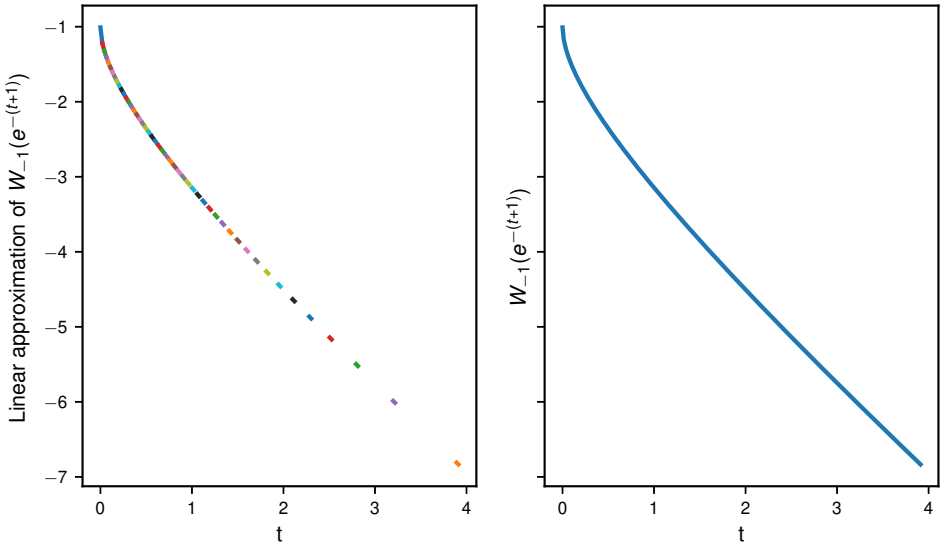

Fig. 14. Lambert function W_{-1} and its linear approximation for $s = 0.02$, $p_{\min} = 0.02$.

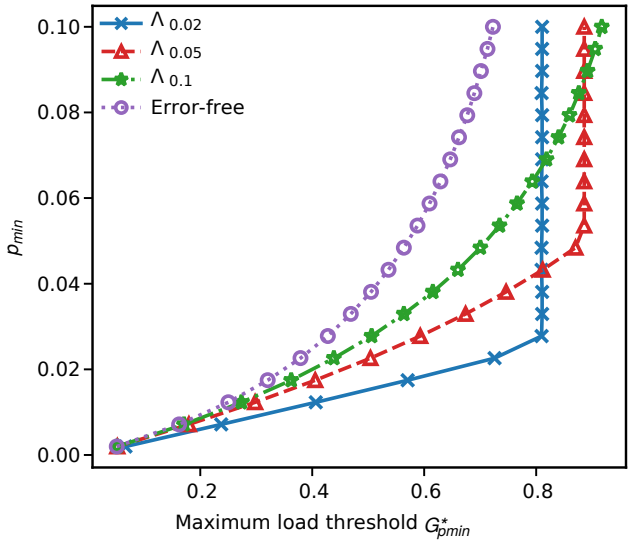

Fig. 15. Performance of different distributions Λ: $p_{\min}$ versus maximum load threshold for $\gamma = 0.99$, $R = 0.4$ and different Λ.

Now by inverting W_0 and rearranging the terms to isolate n on the left hand side we obtain constraint $C4$:

$$\begin{aligned} & \frac{\gamma(p_{\min}-1)}{e^{\gamma}} e^{n-n\gamma} > W_0^{-1}(-\gamma) = -\gamma e^{-\gamma}, \\ \iff \quad & e^{n-n\gamma} < \frac{1}{1-p_{\min}}, \\ \iff \quad & n < -\frac{\ln(1-p_{\min})}{1-\gamma} \triangleq n_{\max}(\gamma, p_{\min}). \end{aligned}$$

B. Lambert Function Approximation

Fig. 14 shows how close are the Lambert function and its approximation, used in Section IV-B.

C. Comparison of Obtained Distributions for Different Loads

To complement the experimental results of Section V, Fig. 15 allows us to better understand the usefulness provided by the different $\Lambda_{p_{\min}}$ distributions. We reverse the x-axis and y-axis and plot $p_{\min}$ as a function of the optimal load threshold for different distributions: $\Lambda_{0.02}$, $\Lambda_{0.05}$, $\Lambda_{0.1}$ and the optimal error-free distribution. We see that the curves increase linearly and slowly until a critical maximum load threshold $\mathcal{G}^*_{\text{crit}}(\Lambda)$, after which the curves become vertical lines. This highlights a *load threshold* $\mathcal{G}^*_{\text{crit}}(\Lambda)$ for all these distributions. If we input a load $G > \mathcal{G}^*_{\text{crit}}(\Lambda)$, then we exceed the load threshold of the distribution and the probability of wrongly decoding a packet $p_{\min}$ dramatically increases. Hence for a fixed distribution $\Lambda_{p_{\min}}$, the maximum load threshold is given by the critical value $\mathcal{G}^*_{\text{crit}}(\Lambda)$, and leads to a probability of wrong decoding $p_{\min}$ and an asymptotic $P_L = \hat{\lambda}(p_{\min})$.

If the distribution is not fixed, the optimal curve $p_{\min} = f(G)$ can be obtained by taking the minimum of all the functions $p_{\min} = f(G)$ for the distributions $\Lambda_{p_{\min}}$. To sum up, the optimization problem helps us to find different distributions $\Lambda_{p_{\min}}$. Each distribution corresponds to a maximal load threshold $G^*_{p_{\min}}$ and its related $p_{\min}$, which can be linked to the PLR P_L. Therefore, there are various trade-offs between the optimal load and the PLR that are achieved by the different distributions. It is interesting to notice that the distributions allowing the highest load threshold also lead to the highest (worst) asymptotic PLR. In particular, the optimal error-free distribution perform poorly in that regard. For instance, if we are going to choose between the four distributions, $\Lambda_{0.02}$ would be preferable if the load remains smaller than 0.8. For a slightly higher load, $\Lambda_{0.05}$ should be chosen. In Section VI, we will confirm by simulations that this phenomenon remains true and is even magnified for a finite number of users and slots.

D. Influence of the Frame Size

To complement the simulation results of Section VI, the influence of the amount of slots, kN, is also studied in Fig. 16 by varying N. We can already see that for $N = 1000$, the PLRs P_L are much more important than in the asymptotic case. This

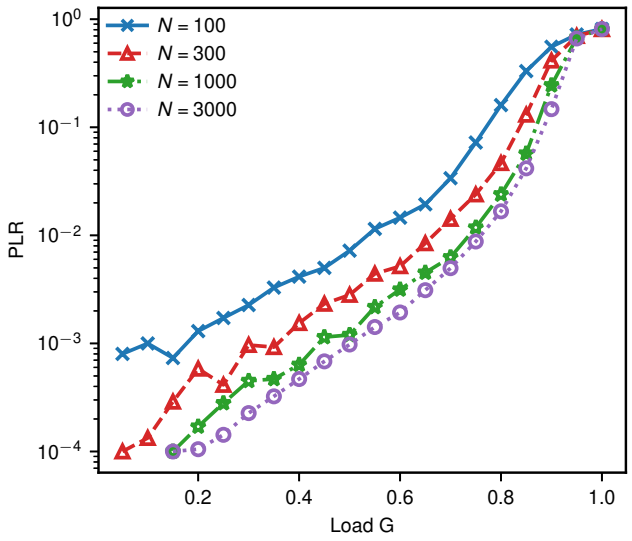

Fig. 16. Influence of N: PLR P_L versus load G for IRSA, $\gamma = 0.99$, optimal error-free distribution and different N.

phenomenon is confirmed here by plotting the PLR for the IRSA optimal error-free distribution: when N increases, the PLR decreases slightly and tends to the asymptotic PLR that was found analytically in Section V. However, even for high N, we are still far from these theoretical PLRs. In practice, the number of slots for a CSA scheme is between 50 [14] and 2000 [37].

REFERENCES

[1] N. Abramson, "Multiple access in wireless digital networks," *Proceedings of the IEEE*, vol. 82, no. 9, pp. 1360–1370, 1994.

[2] C. Morlet, A. B. Alamanac, G. Gallinaro, L. Erup, P. Takats, and A. Ginesi, "Introduction of mobility aspects for DVB-S2/RCS broadband systems," *Space Communications*, vol. 21, no. 1-2, pp. 5–17, 2007.

[3] N. Abramson, "The ALOHA system: Another alternative for computer communications," in *Proceedings of 1970 Fall Joint Computer Conference*, ser. AFIPS, 1970, pp. 281–285.

[4] I. Gitman, "On the capacity of slotted ALOHA networks and some design problems," *IEEE Trans. Commun.*, vol. 23(3), pp. 305–317, 1975.

[5] E. Casini, R. De Gaudenzi, and O. D. R. Herrero, "Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access scheme for satellite access packet networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1408–1419, 2007.

[6] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted ALOHA," *IEEE Trans. Commun.*, vol. 59, no. 2, 2010.

[7] K. R. Narayanan and H. D. Pfister, "Iterative collision resolution for slotted ALOHA: An optimal uncoordinated transmission policy," in *Intl. Symp. on Turbo Codes & Iterative Inf. Process. (ISTC)*, 2012.

[8] M. Luby, M. Mitzenmacher, and M. A. Shokrollahi, "Analysis of random processes via and-or tree evaluation," in *Proc. of the 9th Annual ACM-SIAM Symp. on Discrete Algorithms*, 1998, pp. 364–373.

[9] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb 2001.

[10] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[11] E. Paolini, G. Liva, and M. Chiani, "Coded slotted ALOHA: A graph-based method for uncoordinated multiple access," *IEEE Trans. Inf. Theory*, vol. 61, no. 12, pp. 6815–6832, 2015.

[12] J.-L. Lu, W. Shu, and M.-Y. Wu, "A survey on multipacket reception for wireless random access networks," *Journal of Computer Networks and Commun.*, 2012.

[13] M. Ghanbarinejad and C. Schlegel, "Irregular repetition slotted ALOHA with multiuser detection," in *10th Annual Conf. on Wireless On-demand Network Systems and Services*, 2013, pp. 201–205.

[14] F. Lazaro and Č. Stefanović, "Finite-length analysis of frameless ALOHA with multi-user detection," *IEEE Commun. Lett.*, 2016.

[15] C. Stefanovic, E. Paolini, and G. Liva, "Asymptotic performance of coded slotted ALOHA with multipacket reception," *IEEE Commun. Lett.*, vol. 22, no. 1, p. 2657–2673, Jan. 2018.

[16] I. Hmedoush, C. Adjih, P. Mühlethaler, and V. Kumar, "On the performance of irregular repetition slotted ALOHA with multiple packet reception," in *Intl. Wireless Communications and Mobile Computing Conf. (IWCMC)*, 2020, pp. 557–564.

[17] Z. Sun, Y. Xie, J. Yuan, and T. Yang, "Coded slotted ALOHA for erasure channels: Design and throughput analysis," *IEEE Trans. Commun.*, vol. 65, no. 11, pp. 4817–4830, 2017.

[18] S. Yang, Y. Chen, S. C. Liew, and L. You, "Coding for network-coded slotted ALOHA," in *IEEE Inf. Theory Workshop (ITW)*, 2015, pp. 1–5.

[19] T. Emoto and T. Nozaki, "Shifted coded slotted ALOHA," in *Intl. Symp. on Inf. Theory and Its Applications (ISITA)*, 2018, pp. 291–295.

[20] M. Oinaga, S. Ogata, and K. Ishibashi, "ZigZag Decodable Coded Slotted ALOHA," in *15th Workshop on Positioning, Navigation and Commun. (WPNC)*, 2018, pp. 1–6.

[21] A. G. i Fabregas and Q. Tang, "Coding in the block-erasure channel," in *Australian Commun. Theory Workshop*, 2006, pp. 19–24.

[22] C. R. Srivatsa and C. R. Murthy, "Throughput Analysis of PDMA/IRSA under Practical Channel Estimation," in *IEEE Intl. Workshop on Signal Process. Advances in Wireless Commun. (SPAWC)*, 2019, pp. 1–5.

[23] H.-C. Bui, K. Zidane, J. Lacan, and M.-L. Boucheret, "A Multi-Replica Decoding Technique for Contention Resolution Diversity Slotted ALOHA," in *IEEE Veh. Technol. Conf. (VTC)*, 2015, pp. 1–5.

[24] K. Zidane, J. Lacan, M.-L. Boucheret, C. Poulliat, M. Gineste, D. Roques, C. Bes, and A. Deramecourt, "Effect of Residual Channel Estimation Errors in Random Access Methods for Satellite Communications," in *IEEE Veh. Technol. Conf. (VTC)*, 2015, pp. 1–5.

[25] N. I. Miridakis and D. D. Vergados, "A survey on the successive interference cancellation performance for single-antenna and multiple-antenna OFDM systems," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 312–335, 2013.

[26] S. M. R. Islam, N. Avazov, O. A. Dobre, and K.-s. Kwak, "Power-Domain Non-Orthogonal Multiple Access (NOMA) in 5G Systems: Potentials and Challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 721–742, 2017.

[27] M. Ivanov, F. Brännström, A. G. i. Amat, and P. Popovski, "Error Floor Analysis of Coded Slotted ALOHA Over Packet Erasure Channels," *IEEE Commun. Lett.*, vol. 19, no. 3, Mar. 2015.

[28] F. Clazzer, E. Paolini, I. Mambelli, and Č. Stefanović, "Irregular repetition slotted ALOHA over the Rayleigh block fading channel with capture," in *IEEE ICC*, 2017.

[29] E. E. Khaleghi, C. Adjih, A. Alloum, and P. Mühlethaler, "Near-far effect on coded slotted ALOHA," in *IEEE PIMRC*, 2017.

[30] Č. Stefanović, M. Momoda, and P. Popovski, "Exploiting capture effect in frameless ALOHA for massive wireless random access," in *IEEE WCNC*, 2014, pp. 1762–1767.

[31] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: Model and erasure channel properties," *IEEE Trans. on Information Theory*, vol. 50, no. 11, p. 2657–2673, Sep. 2006.

[32] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 585–598, 2001.

[33] R. M. Corless, G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth, "On the Lambert W function," *Advances in Computational Mathematics*, vol. 5, no. 1, pp. 329–359, 1996.

[34] F. Alzahrani and A. Salem, "Sharp bounds for the Lambert W function," *Integral Transforms & Special Functions*, vol. 29, no. 12, pp. 971–978, 2018.

[35] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, 2016.

[36] P. Popovski, v. Stefanović, J. J. Nielsen, E. de Carvalho, M. Angjelichinoski, K. F. Trillingsgaard, and A.-S. Bana, "Wireless access in ultra-reliable low-latency communication (URLLC)," *IEEE Trans. on Commun.*, vol. 67, no. 8, pp. 5783–5801, 2019.

[37] M. Fereydounian, X. Chen, H. Hassani, and S. S. Bidokhti, "Non-asymptotic coded slotted ALOHA," in *IEEE ISIT*, 2019, pp. 111–115.